



# vRealize Automation 8.1 Release Notes

Updated 19 January 2021

vRealize Automation 8.1 |

- vRA Easy Installer (ISO) build 15996863
- vRA product (appliance) build 15986821

Check for additions and updates to these release notes.

## What's in the Release Notes

- [About vRealize Automation 8.1](#)
- [What's New](#)
- [Before You Begin](#)
- [Known Issues](#)

## New vRealize Automation 8.1 Patch 3

vRealize Automation 8.1 Patch 3 is now available and includes bug fixes in different areas. This is a cumulative update.

For more information and installation instructions, see [KB 79170](#).

### Important

#### Upgrade failure after performing steps in KB 87120

Performing the instructions used to address the CVE-2021-44228 and CVE-2021-45046 log4j vulnerabilities described in KB 87120 can cause upgrade failures for vRealize Automation and vRealize Orchestrator 8.6.2 or earlier. For a workaround, see [KB 87794](#).

## About vRealize Automation 8.1

vRealize Automation 8.1 adds to the vRealize Automation 8.0 capabilities to bring it closer in capability to the vRA 7.x release, reintroducing key capabilities like XaaS and adding capabilities such as AWS GovCloud support and Powershell support in ABX and python, node.js and powershell in vRO.

## What's New

The many benefits of vRealize Automation 8.1 include:

- XaaS
  - [Support for Custom resources based on vRO Types](#)
  - [Custom day2 operations for custom resources and vRealize Automation built-in types](#)
  - [Custom forms import and export CSS](#)
  - [Support complex input types for vRO workflow request in SB and in a form](#)
- CMP
  - [Quick Setup of vSphere endpoints](#)
  - Health badge in vRA by leveraging vROps Integration
  - [Pricing of Private Cloud Deployments \(Rate Cards\)](#)
- Governance/Policy
  - [Approval Policy](#)
  - [Resource limits \(Memory, CPU\)](#)
  - [Storage limit for vSphere - VM templates and Day 0 only](#)
  - [RBAC - OOTB View-Only Roles](#)
  - Native support for SPBM policies
- Polyglot Platform
  - Powershell for ABX
- IaaS Provisioning
  - Persistent Disk as-a-Service (API)
  - AWS GovCloud
  - NSX-T 3.0 Support
  - vSphere 7 Support
  - [Bulk deployment request](#)
  - NSX-V: On-demand Security Group
  - Tagging API improvements
  - [Day 2 Management \(load balancer and network reconfig\)](#)
  - Networking events for subscription (in ABX and vRO)
- Platform
  - Multi-tenancy - Tenant Management
- Ecosystem
  - IPAM SDK
  - Terraform Provider
  - Openshift Support
  - Active Directory Enhancements - Health Status for Active Directory endpoint
- Catalog Content Sources
  - [Catalog content source - OVA from Marketplace](#)
  - [Pipeline as Catalog Items](#)
- Deploy IPv6 workloads on dual-stack IP (IPv4/ IPv6) networks in vSphere
- Support for vRO properties and composite type in vRealize Automation Service broker

VMware vRealize Automation 8.1 includes in-product user assistance

- Use the signpost help to learn about a setting.
- Use the help panel to get more information about a feature or configuration process.

## **Configuration Management with VMware vRealize Automation SaltStack Config**

VMware vRealize Automation SaltStack Config (formerly SaltStack Enterprise) is a configuration management system that maintains virtual machines in defined states. With the addition of

SaltStack Config in vRealize Automation 8.1 release, you can now ensure that specific packages are installed and that any drift from the defined state is quickly addressed. You can also use SaltStack Config to query and execute commands either on individual VMs or groups of VMs at high scale and speed. For information about installing and using the product, see the *SaltStack Config* section of [vRealize Automation 8.1 product documentation](#).

## [Important] Feature and support notices

- If a workflow is imported in Service Broker and has a custom form enabled before enabling Properties and Composite types and the array counterparts were implemented, the custom form needs to be deleted and the workflows imported again in Service Broker to fix the elements in the form.

## New Upgrading to vRealize Automation 8.1

To upgrade vRealize Automation 8.0/8.0.1 to 8.1, refer the [vRealize Automation 8.x upgrade documentation](#). After upgrading vRealize Suite Lifecycle Manager, refer to [KB 75185](#) to map the vRealize Automation 8.1 OVA binary to vRealize Suite Lifecycle Manager.

**Note:** Upgrading from vRA 8.0.1 HF5 to 8.1 GA will fail. Before upgrading, apply [KB 80318](#).

## Migration Assessment in vRealize Automation 8.1

Before you can use the migration assessment service, you must enable it.

1. After upgrading and deploying a new vRealize Automation 8 instance, navigate to Identity and Access Management.
2. Select the user, edit role to be Cloud Administrator and administrator or viewer of the migration service. Add the migration assessment service.
3. Log user out of vRealize Automation 8.
4. Log user in to vRealize Automation 8 to see the Migration Assessment tile.

**Note:** If you had previously run a migration assessment for 8.0 or 8.0.1 you must rerun the assessment on your source environment for 8.1. To rerun the assessment on your source environment for 8.1: navigate to the source instances page, click edit and provide password, select the required tenants and click Next:Assessment.

## Before You Begin

Familiarize yourself with the supporting documents.

- [Install vRealize Automation with vRealize Easy Installer](#)
- [Administering Users in vRealize Automation](#)
- [Using the vRealize Automation 8 Migration Assessment Service](#)

After installing vRealize Automation and setting up your users, you can use the *Getting Started* and *Using and Managing* guides for each of the included services. The *Getting Started* guides include an end-to-end proof of concept. The *Using and Managing* guides provide more in-depth

information that supports your exploration of the available features. Additional information is also available in [vRealize Automation 8.1 product documentation](#).

- [Getting Started with vRealize Automation Cloud Assembly](#)
- [Using and Managing vRealize Automation Cloud Assembly](#)
- [Getting Started with vRealize Automation Code Stream](#)
- [Using and Managing vRealize Automation Code Stream](#)
- [Getting Started with vRealize Automation Service Broker](#)
- [Using and Managing vRealize Automation Service Broker](#)

For information on vRealize Orchestrator 8.1 features and limitations, refer to the [vRealize Orchestrator 8.0 Release Notes](#).

## New Upgrade

vRealize Automation 8.1 supports upgrade from 8.0 by using vRealize Suite Lifecycle Manager.

For information on upgrading to 8.1 using vRealize Suite Lifecycle Manager, see [Upgrade vRealize Automation 8.x with vRealize Suite Lifecycle Manager](#).

Before you upgrade to vRealize Automation 8.1, ensure that the disk space of the root partition and the data partition have at least 20 GB and 48 GB of free disk space, respectively. If it is an HA setup, you need to extend the VM disk in all vRealize Automation VA nodes.

- To increase the vRealize Automation root partition disk to meet the above requirements, increase the vRealize Automation VM "Hard Disk 1" (/dev/sda4) size to 50 GB **WITHOUT POWERING OFF THE VA** in the vCenter server. Then, run the command "vracli disk-mgr resize" on the vRealize Automation VA from the command shell. Disk resize should be reflected within a few minutes.
- To increase the vRealize Automation data partition disk to satisfy above requirements, increase the vRealize Automation VM "Hard Disk 2" (/dev/mapper/data\_vg-data) size to 148 GB **WITHOUT POWERING OFF THE VA** in the vCenter Server. Then, run the command "vracli disk-mgr resize" on the vRealize Automation VA from the command shell. Disk resize should be reflected within few minutes.

**Note:** If a VM snapshot is present, then you cannot expand the vRealize Automation VM disks.

## Resolved Issues

- **Document limitation/workaround for cost estimation using multiple disks(if using count property in blueprint)**  
Currently, Day 0 provisioning of disks with count property is broken as the blueprint UI doesn't generate new syntax for the attached disk in yaml format. As a result, one of the mandatory property of disk cost estimation, i.e. vcUuid, is null and prevents costing estimation for the catalog item.

Workaround: Manually update the syntax of the blueprint in yaml if using count property for disks:

```
attachedDisks: '${map_by(resource.Cloud_Volume_1.id, id =>
{"source":id}
)}}
```

- **Deploying a blueprint with volume attached to a compute instance and an added count property to have multiple disks results in some disks being DETACHED.**

When deploying such BP, post provisioning, for the created deployment (e.g. count: 2), one of the disks always remains DETACHED instead of ATTACHED. Ideally, only the latest syntax (`map_to_object(resource.disk[*].id)`), in case of multiple disks as the value for "attachedDisks" property, should be allowed. Cost estimation is also not supported in the catalog UI and results in error if such blueprint gets published as a catalog.

Workaround: Add the count property with required number of disks and then only create a link between disk and machine in the blueprint canvas. This method ensures yaml always gets the latest syntax for attachedDisks property. Otherwise, you must manually update to a newer syntax when using count property for adding multiple volumes, once the disk is attached with a compute instance. The correct syntax to be updated in blueprint manually is: `attachedDisks: '${map_by(resource.Cloud_Volume_XYZ.id, id => {"source":id})}'`

- **ABX might not work if Internet access sits behind a Proxy**

ABX actions are executed in prepared on the fly containers that run inside the vRA appliances.

The preparation of these containers require automatic downloads of artifacts that are available in public repositories as industry standard delivery mechanism.

vRA deployments that would like to take benefit of ABX actions should be assigned to virtual networks that have open access to such repositories. Identical network configurations are required for all 3 nodes, when vRA is deployed in cluster.

An example of standard repositories that should be accessible via direct internet access or through a proxy:

For all actions: <https://symphony-docker-external.jfrog.io> & <https://gcr.io> & <https://hub.docker.com/>

For Python actions: <https://pypi.org/>

For NodeJS actions: <https://registry.npmjs.org/>

Opening access to additional repositories might be also required based on the actual dependencies of the ABX actions.

These requirements also apply to the default IPAM and AD provisioning configuration in vRA, which is backed by ABX actions.

Workaround: Use a HTTP proxy to pass traffic to the required external sites. It is configured through **vracll proxy** command line extension and the additional instructions can be obtained via GSS.

- **Cannot set wildcard certs for certain domain names, specifically those not using a Public Suffix.**

vRealize Automation 8 supports setting a wildcard certificate only for DNS names that match the content of the Public Suffix List ([<https://publicsuffix.org/>]) For example, a valid wildcard certificate: you can use a wildcard certificate with DNS name like "\*.myorg.com". This is supported because "com" is part of the Public Suffix List. An invalid wildcard certificate example: you cannot use a wildcard certificate with DNS name like "\*.myorg.local". This is not supported because "local" is not part of Public Suffix List.

Workaround: Only use domain names in the Public Suffix List.

- **Directed to Cloud.vmware.com for access**

The No access error page is shown for logged in user with rights in organization. This only occurs in HA.

Workaround: Clear browser cache.

- **vRA 8 cluster fails to start after VAs have been reverted to a snapshot**

Snapshot of 3-node vRealize Automation 8 cluster from within LCM is currently unavailable.

Workaround: Shutdown vRA services before taking an offline snapshot

1. Run `"/opt/scripts/deploy.sh --onlyClean"` on a single vRA node to shutdown the services safely.
2. Power off each node using the halt command.
3. Take a snapshot after the vms are powered off.

Startup procedure when environment is reverted to snapshots:

1. Power up all vms.
2. Run the `"deploy.sh"` script with no arguments for vRA services to come back up.

- **Provisioning fails with EBS topic not registered after stopping the primary db node**

In a vRealize Automation 8 HA environment, after removing the primary db node, provisioning fails with following error: "Failed to publish event as EBS topics are not registered".

Workaround: See KB for more information.

- **Link to Migration Guide on the Migration Assessment Getting Started page is invalid**

The link to the migration guide in the Migration Assessment UI is incorrect and invalid,

Workaround: The correct link is [Using the vRealize Automation 8 Migration Assessment Service](#).

- **vRO workflow with input from type 'properties' can not be triggered.**

If a vRealize Orchestrator workflow, with input from type properties is exposed in catalogSteps, and is then triggered from vRealize Automation catalogResult, the run fails.

## Known Issues

The following known issues are present in this release.

- **vRA 8.1 Deployment or upgrade fails when the appliance is deployed in a 172.17.x.x network**

vRA deployment fails - deploy.sh script failure at the "Registering embedded vRO" stage  
/var/log/deploy.log contains:

```
curl: (22) The requested URL returned error: 400 Bad Request
Failed to register vRO. Will retry in 45 seconds...
```

...

```
curl: (22) The requested URL returned error: 400 Bad Request
Maximum number of retries exceeded."
```

Cause: The appliance got an IP address from the 172.17.x.x space. This conflicts with in internal *docker0* interface from the vRO pod

Refer to <https://kb.vmware.com/s/article/78783>

- **When a vCenter cloud account is updated to add a data center, the resources from this data center are not immediately available for use.**

Changes made to regions (data centers) for a vCenter cloud account do not take immediate effect and require data collection to run.

Workaround: Wait for the next data collection to complete successfully. Data collection runs approximately every 10 minutes.

- **PowerShell tasks appear to be stuck**

When there is no active session PowerShell tasks appear to be stuck. This behaviour is seen because the PowerShell process responsible to run the user script is held by windows system process WmiPrvSE.

Workaround : Login to the system and keep an active session. Lock the screen instead of completely logging out.

- **vRO represents Array types as complex types with only one column, rather than a field whose "type.isMultiple" is true.**

When adding a workflow which has an array input and consequently customizing its form, do not change the ID of the column in the Values tab of the data grid. The default value must stay set at `_column-0_` . Conversely, you can change the label of the column (which is visible in the UI when adding values to the datagrid).

- **License re-configuring is not supported.**

After configuring vRealize Automation with the Enterprise license, the system can not be re-configured to use the Advanced License.

- **vRealize Automation 8 does not support Internet Explorer 11**

You cannot use Internet Explorer 11 with vRealize Automation 8.

Workaround: Use a different browser instead of Internet Explorer 11.

- **BP Canvas is not refreshed after custom resource has been changed or deleted.**

If you delete a custom resource, the change is not propagated to the Blueprint canvas immediately.

The Canvas has a cache mechanism, which can be updated after using refresh button, next to the search pane.

- **Create different custom resources with the same vRO object type is not supported**

In vRA 7.X it was possible to create different custom resources for the same type. This allowed users to define a different set of create / delete / operate actions for the same vRO type with creating different custom resource types. In vRA 8.1 We do not support a case where same vRO\_Type can be leveraged from different custom resources.

- **vRO workflow is not executed through catalog when there is empty input with reference type**

Null pointer exception appears on attempt to request vRO Workflow with an empty value for the Workflow input with a reference type.

Workaround: Set a default value for the reference type or make the field mandatory.

- **Unsuccessfully provisioned custom resource can't be deleted from a deployment**

When you request a custom resource, if the workflow run that creates the resource fails, a resource in the deployment service is still created (since we are replying to the initial request with a STARTED status which in turn creates the resource in deployment). This resource cannot be deleted since it doesn't contain the metadata that is added upon successful provisioning of the resource in vRO.

Workaround: Right after the first attempt to delete the custom resource, a dialog appears which asks you whether you want to force deletion. Say yes to force its deletion.

- **Custom Resource Name is not propagated correctly to the deployment view list**

When you create a custom resource based on vRO\_Type, you usually use a comprehensive display name. Currently this display name is not available in the Deployment view. The resource, which appears in the deployment is identified only by its type.

- **Available option to set timezone from vCenter Machine Console window**

Undefined behaviour when user sets timezone from vCenter Machine Console window

Workaround: Don't change the time zone.

- **Tenant Names with different cases are treated the same way**

A tenant named vmware and another one named VMware are seen as the same.

Workaround: Tenants in vRA 8.1 are based on hostnames since hostnames are case insensitive the tenant names are also case insensitive. This means that a tenant named

VMware is the same as VMWARE or vmware or any other combination cases. The tenant name capitalization may vary and may not be preserved across the application.

- **Blueprints with property bindings to certain networking properties fail to deploy because the binding values cannot be resolved correctly.**

Property bindings for the **dns**, **dnsSearchDomains**, and **gateway** properties are not working. These are primarily used with OVF blueprints.

Workaround: Blueprints using the following properties must be modified to use a different set of properties.

**Note:** A permanent fix for this issue will be delivered in the first hotfix for vRA 8.1. The workaround provided here should be considered temporary and will need to be reverted after the hotfix is applied.

For the **dns** property:

```
dns0: '${resource.Cloud_NSX_Network_1.dns[0]}'  
dns1: '${resource.Cloud_NSX_Network_1.dns[1]}'
```

*must be changed to*

```
dns0: '${replace(split(resource.Cloud_NSX_Network_1.dnsServerAddresses, ",")[0], "[", "")}'  
dns1: '${replace(split(resource.Cloud_NSX_Network_1.dnsServerAddresses, ",")[1], "[", "")}'
```

For the **dnsSearchDomain** property:

```
dnsSearchDomain0: '${resource.Cloud_NSX_Network_1.dnsSearchDomains[0]}'  
dnsSearchDomain1: '${resource.Cloud_NSX_Network_1.dnsSearchDomains[1]}'
```

*must be changed to*

```
dnsSearchDomain0: '${replace(split(resource.Cloud_NSX_Network_1.dnsSearchDomains, ",")[0], "[", "")}'  
dnsSearchDomain1: '${replace(split(resource.Cloud_NSX_Network_1.dnsSearchDomains, ",")[1], "[", "")}'
```

For the **gateway** property:

```
gateway: '${resource.Cloud_NSX_Network_1.gateway}'
```

*must be changed to*

```
gateway: '${resource.Cloud_NSX_Network_1.gatewayAddress}'
```

- **Node's CPU usage jumps to 100%, pods start crashing**

When trying to generate a log bundle on a highly-loaded environment, it is possible to temporarily overload one or more of its nodes in terms of CPU and/or memory usage. This may cause services to crash.

Workaround: Run the log bundle collection script when the environment is not loaded. Configure and monitor log forwarding to an external logging solution (vRLI or syslog server).

- **Data collection fails to collect storage policies, and fails to update existing storage**

**policies with compatible datastores or vCenter 7.0. Data collection fails to update WCP availability in vRA.**

If there are multiple datacenters in a vSphere cloud account and not selected in vRA's endpoint, this could cause failures towards completing the data collection and the data collection is partially successful and causes above symptoms.

Workaround: Select all datacenters (regions) in vSphere cloud account. If there is no intention to manage that datacenter, you don't need to create the cloud zone. However the datacenter's artifacts will be collected.

- **Binding of Custom day2 action and built-in type needs to be manually referenced**

In vRA 7.X there was an automating binding of Custom day2 action and in context vRA built-in object. In vRA 8.1 this binding should be made via vRO action.

You can check the official documentation for more guidance on the binding process.

- **When a deployment is missing a resource and the user tries to update the deployment by applying a blueprint in generation of plan, the user might see the "Another request is already in progress on deployment" error message.**

The user will also see an additional "Day 2 Action - Delete" in the deployment history timeline. Also, when user tries to update the deployment via API they see "Another request in progress on deployment".

Retry updating the deployment.

- **When importing a vRO workflow as XaaS catalog item that has actions which populate dropdowns, selectable values are imported as static constants**

When importing a vRO workflow as XaaS catalog item that has actions which populate dropdowns, selectable values are imported as static constants.

This means that when the user requests the catalog item, the request form is presented with static values rather than dynamically populated fields.

For such catalog items use custom forms and manually select "external source" and browsing action which will populated the value correctly.

- **NEW vRO Workflow presentation with an OGNL expression does not render properly when used as a custom day2 operation in vRA.**

Custom Resource Actions with workflows that have OGNL constraints in their presentation may not render properly and it may not be possible to populate all required fields.

- **NEW When filtering a list of load balancers by name, the same vRA-deployed NSX load balancer appears twice with slightly different names - once as "Deployed" and again as "Discovered".**

When vRA deploys an NSX load balancer, the load balancer is created in NSX using a different ID and name than vRA uses in its internal database. As a result, vRA creates and subsequently updates a new, duplicate load balancer record when it data collects the associated NSX cloud account instead of updating the load balancer record it originally created. This results in the confusing appearance of near-duplicate pairs of load balancers in screens where load balancers are listed.

Workaround: When adding a vRA-deployed NSX load balancer to a network profile, select the one that is "Deployed" instead of the one that is "Discovered".

Copyright © 2023 VMware, Inc. All rights reserved.